#include<stdio.h>

#include<conio.h>

#include<alloc.h>

struct node

{

int info;

struct node \*next;

};

typedef struct node node;

node\* create();

node\* insbeg(node\*);

node\* insmid(node\*);

node\* insend(node\*);

int count(node\*);

void display(node\*);

void main()

{

int choice;

node \*head;

head=NULL;

while(1)

{

clrscr();

printf("\n\n\t\tThis is the program of LINEAR LINKED LIST.");

printf("\n\n\n\n\t\tWhich operation you want to perform --");

printf(“\n\t\t 1. Create the list.”)

printf("\n\t\t 2. Insert element at the beginning OR");

printf("\n\t\t 3. Insert element at the middle OR");

printf("\n\t\t 4. Insert element at the end OR");

printf("\n\t\t 5. Count the number of elements OR");

printf("\n\t\t 6. Display the list OR");

printf("\n\t\t 7. Wanna exit from the program.");

printf("\n\n\n\n\t\tEnter your choice = ");

scanf("%d",&choice);

switch(choice)

{

case 1: head = create();

continue;

case 2: head=insbeg(head);

continue;

case 3: head=insmid(head);

continue;

case 4: head=insend(head);

continue;

case 5: clrscr();

printf("\n\n\t\tThe list has %d elements.",count(head));

printf("\n\n\t\t\*\*Press any key to continue.\*\*");

getch();

continue;

case 6: display(head);

continue;

case 7:printf("\n\n\t\*\*successfully terminated from the program.\*\*");

getch();

exit(0);

default:printf("\n\n\t\tYou entered a invalid choice!!!");

printf("\n\n\t\tPress any key to choose again.......");

getch();

}

}

}

node \* create()

{

node \*p, \*temp,\*head;

head = (node\*)malloc(sizeof(node));

head->next=NULL;

printf(“\n\n\t Enter data to insert or -999 to stop”);

scanf(“%d”, &head->info);

p=head;

while(p->info!= -999)

{

temp=(node\*)malloc(sizeof(node));

temp->next=NULL;

scanf(“%d”,temp->info);

p->next=temp;

p=temp;

}

return(head);

}

node\* insbeg(node \*head)

{

node \*ins;

int num;

clrscr();

printf("\n\n\t\tEnter the number to insert at the beginning = ");

scanf("%d",&num);

ins=(node\*)malloc(sizeof(node));

ins->info=num;

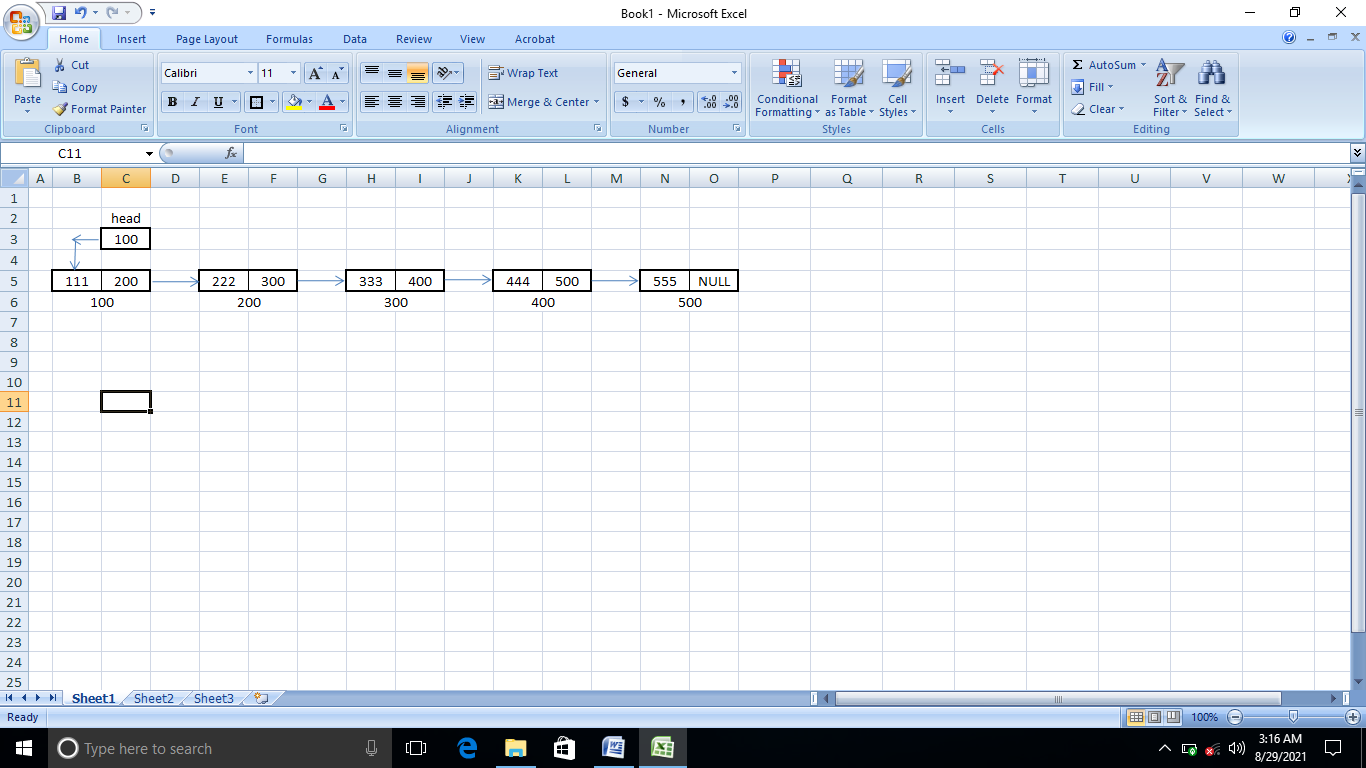
ins->next=head;

printf("\n\n\t\t\*\*Your element has been successfully inserted.\*\*");

getch();

return ins;

}



node\* insmid(node \*head)

{

node \*tmp,\*ins;

int pos,cout,num,i;

cout=count(head);

while(1)

{

clrscr();

printf("\n\n\t\tAt which position you want to insert the element = ");

fflush(stdin);

scanf("%d",&pos);

if(pos>cout+1 || pos<1)

{

printf("\n\n\t\tYou entered a wrong position.");

printf("\n\n\t\tPress any key to choose any other position.");

getch();

continue;

}

else

{

ins=(node\*)malloc(sizeof(node));

printf("\n\n\n\n\t\tEnter the number to insert = ");

fflush(stdin);

scanf("%d",&num);

if(pos==1)

{

ins->info=num;

ins->next=head;

return ins;

}

else

{

tmp=head;

for(i=2;i<pos;i++,tmp=tmp->next);

ins->info=num;

ins->next=tmp->next;

tmp->next=ins;

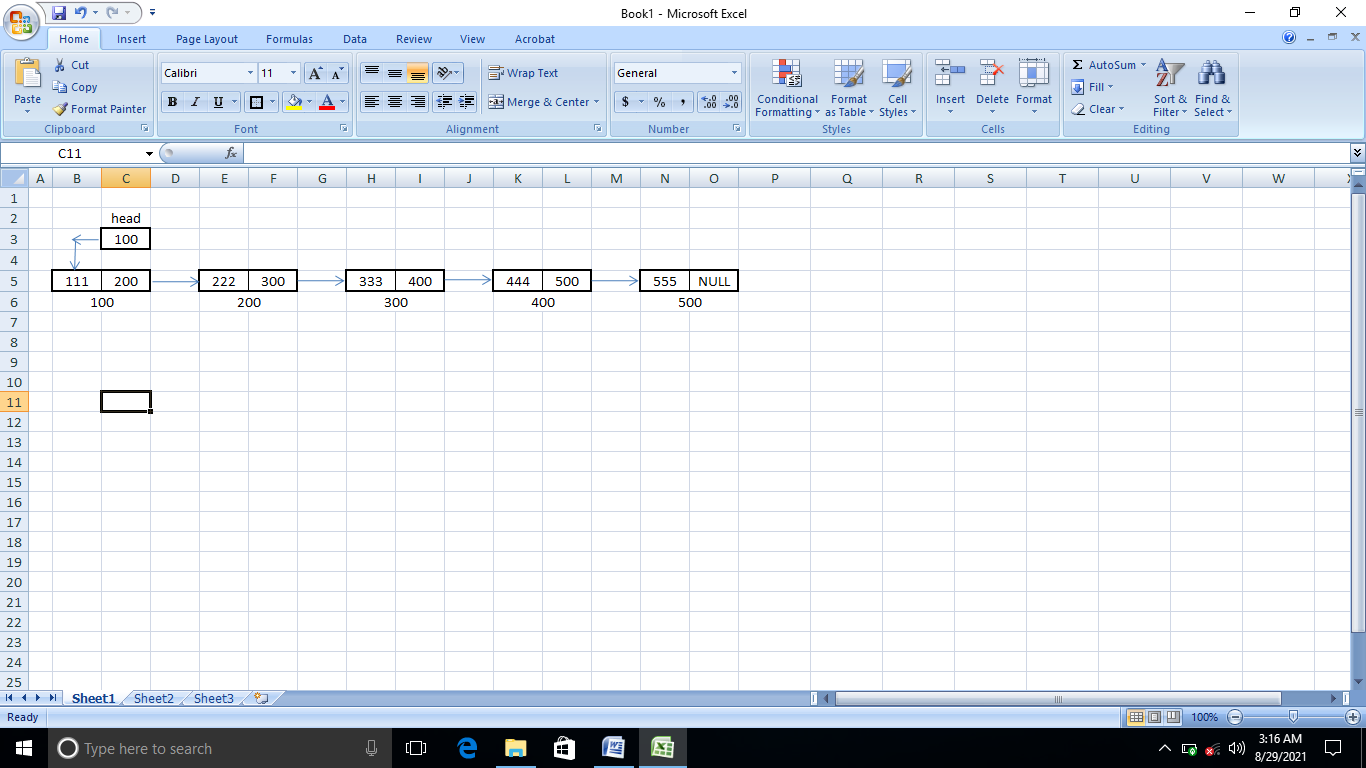
return head;

}

}

}

}



node\* insend(node \*head)

{

node \*ins,\*tmp;

int num;

clrscr();

printf("\n\n\t\tEnter any number to insert at the end = ");

fflush(stdin);

scanf("%d",&num);

if(head==NULL)

{

head=(node\*)malloc(sizeof(node));

head->info=num;

head->next=NULL;

}

else

{

for(tmp=head;tmp->next!=NULL;tmp=tmp->next);

ins=(node\*)malloc(sizeof(node));

tmp->next=ins;

ins->info=num;

ins->next=NULL;

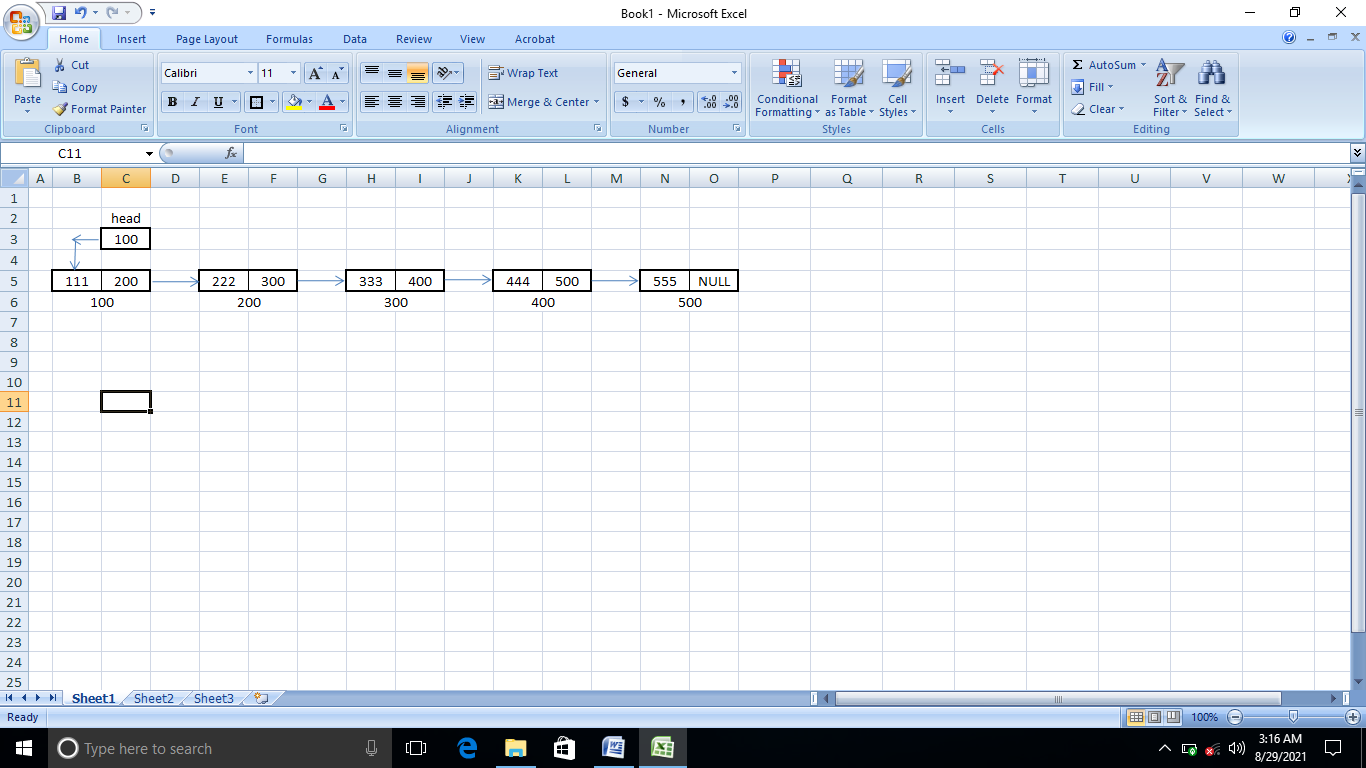
}

printf("\n\n\t\t\*\*Your element has been successfully inserted at end.\*\*");

getch();

return head;

}



int count(node \*head)

{

int count;

for(count=0;head!=NULL;count++,head=head->next);

return count;

}

void display(node \*head)

{

node \*tmp;

tmp=head;

clrscr();

if(head==NULL)

{

printf("\n\n\t\tThere is no element in the linked list.");

printf("\n\n\t\tPlease inseart some elements first then try again.");

}

else

{

printf("\n\n\t\tYour list is as follows --\n");

while(tmp!=NULL)

{

printf("\n\t\t%d",tmp->info);

tmp=tmp->next;

}

}

getch();

}